CLAIMS

What is claimed is:

1. A full-text search system comprising:

a plug-in component that defines a relevant score algorithm; and

a search component adapted to receive and utilize the plug-in component to query data and generate results.

2. The system of claim 1, wherein the plug-in component defines a full-text index schema.

3. The system of claim 2, wherein the plug-in component specifies how the schema is to be populated.

4. The system of claim 1, wherein the search component is tightly integrated into a database management system.

5. The system of claim 4, wherein the database management system query execution engine executes received full-text queries and database queries.

6. The system of claim 5, wherein the full-text queries are optimized by the database management system optimization component.

7. The system of claim 1, wherein the search component comprises an index system that creates an index in accordance with one or more consumer plug-in components and a query processing system that provides a mechanism for retrieving relevant documents utilizing the index.

8. The system of claim 7, wherein the index is a compressed nested data structure.

9. A full-text indexing system comprising:

a gatherer component to retrieve a document from a data store;

a producer pipeline component that parses the structure and text of the retrieved document in accordance with a plurality of third party developer specified components; and

a consumer pipeline component that receives data from the producer pipeline component and persists data to an inverted index.

10. The system of claim 9, wherein the gather component and the consumer pipeline component reside within a database management system.

11. The system of claim 10, wherein the producer pipeline component is executed as an external daemon process managed by an external host controller component residing within the database management system.

12. The system of claim 9, wherein the producer pipeline component comprises a noise component to remove keywords that are diminutive in value as search criteria.

13. The system of claim 9, wherein the gatherer component retrieves documents from external databases.

14. A full text query system tightly integrated with a database management system comprising:

a parser component that tokenizes received queries;

an execution plan generation system that generates an execution plan based on tokens received from the parser component and a ranking algorithm provided by a third party developer *via* a ranking plug-in component;

an execution engine component that utilizes the execution plan to search an index and produce query results in order as specified by the ranking algorithm.

15. The system of claim 14, further comprising a user interface component to receive queries from users.

16. The system of claim 14, wherein execution plan generator component produces a query tree, appends scoring functions to leaves, and transforms the query tree into a data base query expression.

17. The system of claim 16, further comprising an optimizer component that optimizes the database query structure based on information concerning the index to be searched.

18. The system of claim 17, further comprising an expander component that modifies keywords provided in the query structure.

19. The system of claim 18, wherein the expander component is executed as a separate daemon process from the execution engine.

20. The system of claim 19, wherein the expander component includes at least one of a stemmer component, a normalizer component, an inflection component, a thesaurus component, a custom expansion component, a homonym component, and a fuzzy component.

21. The system of claim 20, where the expander components and associated functionality are specified by a third party developer.

22. A method of employing a customized full-text query comprising:

retrieving a full-text indexing schema and ranking algorithm from a plug-in component provided by a third party developer; and

populating an index in accordance with the provided indexing schema.

23. The method of claim 22, further comprising:

receiving a query;

generating results utilizing the index; and

displaying the results by rank in accordance with the ranking algorithm, wherein the results are displayed in order from most to least relevant.

24. The method of claim 23, wherein the results are generated by a database management system query processor.

25. A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 22.

26. A customized indexing methodology comprising:

retrieving a document from a data source;

removing document formatting data and emitting text chunks;

parsing the text chucks into keywords; and

persisting the keywords to an index, the index schema being defined by a third party developer.

27. The method of claim 26, further comprising normalizing keywords.

28. The method of claim 26, further comprising removing keywords of diminutive value as search criteria.

29. The method of claim 26, further comprising identifying the language of each text chunk and generating an id indicative thereof.

30. The method of claim 26, further comprising determining and noting the position of each keyword within a respective document.

31. A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 26.

32. A full text search methodology comprising:

receiving a search request; and

generating a query expression in response to the search request which includes a third party developer specified ranking algorithm for determining the relevance of result documents.

33. The method of claim 32, further comprising utilizing a database management system to optimize the query expression prior to execution.

34. The method of claim 33, further comprising employing a database execution engine to execute full text queries.

35. The method of claim 32, further comprising modifying the query expression to include, remove, add, or modify keyword terms.

36. The method of claim 35, wherein the query expression is modified by one or more components specified by a third party developer.

37. The method of claim 36, wherein the developer specified components are executed as separate daemon processes managed by an external host controller component from within a database management system.

38. The method of claim 35, wherein the query expression is modified once at compile time and again at runtime.

39. The method of claim 32, further comprising displaying search results in relevant order to a user.

40. A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 32.